

Universidad Tecnológica Nacional
Facultad Regional Avellaneda



Técnico Superior en Programación

Técnico Superior en Sistemas Informáticos

Materia: Laboratorio de computación I

Apellido:					Fecha:	22/11/2018				
Nombre:					Docente:					
División:					Nota:					
Legajo:					Firma:					
Instancia	PP		RPP		SP	X	RSP		FIN	

PARTE 1

Realizar un programa que permita analizar información de ventas de productos almacenadas en un archivo. El programa deberá iniciar y cargar en una lista los elementos definidos en el archivo data.csv, el cual cuenta con el siguiente formato:

ID_Venta;Fecha_Venta;Codigo_Producto;Cantidad;Precio_Unitario;CUIT_Cliente

Por ejemplo:

33;16/11/2018;LCD_TV;1;12567.99;20-33263109-4

34;17/11/2018;MOUSE_598;3;123.99;21-33782109-5

Luego deberá generar un archivo "informes.txt" con el siguiente formato:

Informe de ventas

- Cantidad de unidades vendidas totales: XX
- Cantidad de ventas por un monto mayor a \$10000: XX
- Cantidad de ventas por un monto mayor a \$20000: XX
- Cantidad de TVs LCD vendidas: XX

Requerimientos del desarrollo.

- Se deberá crear la entidad "Venta" con todos sus campos correspondientes.
- Se deberá utilizar la biblioteca LinkedList para almacenar las ventas leídas del archivo.
- Se deberá agregar a la biblioteca la función "Il_count()" la cual permitirá contar elementos de la lista bajo algún criterio.
- Se deberá utilizar la función "Il_count()" para calcular los informes pedidos.

Detalle de la función "ll_count"

Prototipo de la función:

```
int ll_count(LinkedList* this, int (*fn)(void* element))
```

La función "ll_count" recibirá una lista y una función "fn". Se deberá iterar todos los elementos de la lista y pasárselos a la función "fn". La función "fn" devolverá la cantidad que debe contarse. La función "ll_count" almacenará un acumulador al cual sumará el valor de retorno de "fn" en cada iteración. Al finalizar las iteraciones, la función "ll_count" devolverá el valor acumulado.

Nota 0: El código deberá tener comentarios con la documentación de cada una de las funciones y respetar las reglas de estilo de la cátedra.

Nota 1: Separar en archivos las entidades, parser y generador de informes.

PARTE 2

Preguntas oral:

- Estructura de la LinkedList.
- Función count.
- Otra función de la LinkedList.

CONDICIONES DE APROBACIÓN

Para la aprobación directa (nota ≥ 6), se deberá tener el programa funcionando en su totalidad como se pide en la parte 1 del examen y haber contestado todas las preguntas de la parte 2.

Para la aprobación con final (nota = 4 o 5), se deberá realizar el parseo del archivo, la función ll_count, las funciones para contar de al menos 2 informes y haber contestado la pregunta oral de la estructura de la LinkedList.